**安装包**

react-native-webview

**简单示例**

import React from 'react'

import WebView from 'react-native-webview'

class Component extends React.Component {

    createHtml() {

        return `

        <div id="canvas" style=""></div>

        <script>

            window.onload = function () {

            }

        </script>

        `

}

    render() {

        return (

            <WebView

                source={{ html: this.createHtml() }}

            />

        );

    }

}

export default Component;

**高度自适应**

import React from 'react'

import WebView from 'react-native-webview'

interface Props {

    html: string

}

export default class IEWebView extends React.Component<Props> {

    state = {

        webHeight: 0

}

    constructor(props) {

        super(props);

}

    createHtml(html) {

        return `

        <body style="margin: 0px">

            <div id="\_\_container\_\_">

                ${html}

                <div></div>

            </div>

            <script>

                var oldonload = window.onload;//得到上一个onload事件的函数

                window.onload = function () {

                    if (typeof oldonload == 'function') {

                        oldonload();

                    }

                    setTimeout(()=>{

                        // 向 WebView 发送消息

                        let webHeight = document.getElementById("\_\_container\_\_").clientHeight;

                        window.ReactNativeWebView.postMessage(webHeight);

                    }, 1)

                }

            </script>

        <body>

        `

}

    onMessage = (msg) => {

        console.log("--------------------------")

        console.log(msg.nativeEvent)

        if (msg.nativeEvent.data !== undefined && msg.nativeEvent.data !== null) {

            let height = parseInt(msg.nativeEvent.data);

            if (isNaN(height)) {

                height = 0;

            }

            this.setState({

                webHeight: height

            });

        }

}

    render() {

        console.log(this.createHtml(this.props.html));

        return (

            <WebView

                style={{ width: '100%', height: this.state.webHeight }}

                javaScriptEnabled={true}

                source={{ html: this.createHtml(this.props.html) }}

                onMessage={this.onMessage}

                scalesPageToFit={false}

            />

        );

    }

}